Acturary   Norrect   N	VERSION 1: raw	talk.religion.misc 238 251	
Alt athelism	Category NCorrect N	0.948	
Accuracy   Comp.graphics   S9   389   alt.atheism   S3   319	Accuracy	VERSION 2: mest	
Comp.graphics   59   389	alt.atheism 26 319	Category NCorrect N	
0.152       0.166         comp.os.ms-windows.misc       33       394       comp.graphics       161       389       389         0.084       0.084       0.144       0.089.sys.jbm.pc.hardware       47       392       0.12       comp.sys.mac.hardware.misc       54       394       394         0.025       comp.syna.chardware       48       385       0.137       0.137       0.126       392       0.22       0.321       0.137       0.137       0.148       0.148       0.321       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.322       0.328       0.328       0.328       0.328       0.328       0.328       0.328 <t< td=""><td>0.082</td><td>Accuracy</td><td></td></t<>	0.082	Accuracy	
Comp. os.ms-windows.misc	comp.graphics 59 389	alt.atheism 53 319	
0.084	0.152	0.166	
Comp. sys. ibm. pc. hardware	comp.os.ms-windows.misc 33 394	comp.graphics 161 389	
comp.sys.mac.hardware         48         385         0.137           0.125         comp.sys.ibm.pc.hardware         126         392           comp.sys.ibm.pc.hardware         126         392	0.084	0.414	
0.125     comp.sys.ibm.pc.hardware     126     392	comp.sys.ibm.pc.hardware 47 392 0.12	comp.os.ms-windows.misc 54 394	
comp.windows.x         58         392         0.321           0.148         comp.sys.mac.hardware         122         385           misc.forsale         67         390         0.317           0.172         comp.windows.x         183         392           rec.autos         0.467         0.467           0.104         misc.forsale         122         390           rec.motorcycles         38         398         0.313           0.095         rec.autos         150         395         0.38           rec.sport.baseball         30         397         rec.motorcycles         147         398         0.38           0.076         0.369         rec.sport.baseball         111         397         0.28           rec.sport.hockey         49         399         rec.sport.baseball         111         397         0.28           0.123         rec.sport.hockey         146         339         0.28           sci.crypt         41         396         0.366         396         0.359           0.056         sci.electronics         76         393         0.359           sci.space         36         394         0.133         0.297         11	comp.sys.mac.hardware 48 385	0.137	
0.148     comp.sys.mac.hardware     122     385	0.125	comp.sys.ibm.pc.hardware 126 392	
Misc.forsale	comp.windows.x 58 392	0.321	
Comp.windows.x	0.148	comp.sys.mac.hardware 122 385	
Tec. autos	misc.forsale 67 390	0.317	
0.104 rec.motorcycles	0.172	comp.windows.x 183 392	
No.095	rec.autos 41 395	0.467	
The composition of the composi	0.104	misc.forsale 122 390	
rec.sport.baseball	rec.motorcycles 38 398	0.313	
0.076       0.369         rec.sport.hockey	0.095	rec.autos 150 395 0.38	
rec.sport.hockey	rec.sport.baseball 30 397	rec.motorcycles 147 398	
0.123       rec.sport.hockey	0.076	0.369	
sci.crypt	rec.sport.hockey 49 399	rec.sport.baseball 111 397 0.28	
0.104       sci.crypt	0.123	rec.sport.hockey 146 399	
sci.electronics	sci.crypt 41 396		
0.056       sci.electronics       76       393          sci.med       30       396        0.193        396          sci.space       36        0.313        394        394         397        0.297        398         113        398         113        398         0.297         124        394	0.104	sci.crypt 142 396	
sci.med       30       396       0.193         0.076       sci.med       124       396          sci.space       124       396          sci.space       117       394          sci.space       117       394          sci.space       117       394          sci.space       117       394          sci.space       117       398          sci.space       117       398          0.297       sci.space       113        398          talk.politics.guns       30        398        0.284        0.284        0.22        364        0.22        376        0.202	sci.electronics 22 393	0.359	
0.076       sci.med	0.056	sci.electronics 76 393	
sci.space	sci.med 30 396	0.193	
0.091       sci.space	0.076	sci.med 124 396	
soc.religion.christian	sci.space 36 394	0.313	
0.118 soc.religion.christian 113 398 talk.politics.guns 32 364 0.284  0.088 talk.politics.mideast 26 376 talk.politics.mideast 76 376 0.069 0.202 talk.politics.misc 14 310 talk.politics.misc 36 310	0.091	sci.space 117 394	
talk.politics.guns	soc.religion.christian 47 398	0.297	
0.088 talk.politics.guns	0.118	soc.religion.christian 113 398	
talk.politics.mideast	talk.politics.guns 32 364		
0.069 0.202 talk.politics.misc 14 310 talk.politics.misc 36 310	0.088	talk.politics.guns 80 364 0.22	
talk.politics.misc 34 310 talk.politics.misc 36 310	talk.politics.mideast 26 376	talk.politics.mideast 76 376	
	0.069	0.202	
0.045	talk.politics.misc 14 310	talk.politics.misc 36 310	
	0.045	0.116	

talk.religion.misc 0.821	206	251	
VERSION 3: tfidf Category	NCorrect	N	
Accuracy alt.atheism 0.295	94	319	
comp.graphics	205	389	
comp.os.ms-windows.misc 0.472	186	394	
comp.sys.ibm.pc.hardware	172	392	
comp.sys.mac.hardware 0.608	234	385	
comp.windows.x	160	392	
misc.forsale	249	390	
rec.autos	201	395	
rec.motorcycles rec.sport.baseball 0.436			 0.58
rec.sport.hockey 0.429	171	399	
sci.crypt	164	396	
sci.electronics	145	393	
sci.med	147	396	
sci.space	136	394	
soc.religion.christian 0.291	116	398	
talk.politics.guns 0.283	103	364	
talk.politics.mideast 0.186	70	376	

```
talk.politics.misc ----- 53 ----- 310 ---- 0.171 talk.religion.misc ----- 223 ----- 251 ---- 0.888
```

The prior data was acquired through running each of the versions in my program. In the following explanation use the pursuing legend for comprehension.

```
v<sub>j</sub> = specific category
w<sub>k</sub> = specific word
```

## **Version 1: Raw**

The results from the raw version were very poor, with an average unweighted accuracy of 14.4%. Ignoring talk.religion.misc as an outlier, the average becomes 10.5%. The lowest accuracy for a given category was 4.5% for talk.politics.misc. The highest accuracy for a given category was 94.8% for talk.religion.misc.

## **Version 2: M-Est**

The results from the M-Estimate version improved upon the results from the raw version. The unweighted accuracy was 31.7%. Ignoring talk.religion.misc as an outlier, the average becomes 29%. The lowest accuracy for a given category was 11.6% for talk.politics.misc. The highest accuracy for a given category was 82.1% for talk.religion.misc.

## Version 3: tf-idf

The results from the tf-idf version were by far the best. The unweighted accuracy average was 43.3%. Ignoring talk.religion.misc as an outlier, the average becomes 40.9%. The lowest accuracy for a given category was 17.1% for talk.politics.misc. The highest accuracy for a given category was 88.8% for talk.religion.misc.

## Results analysis

The reason why M-Estimate improved upon raw is due to  $P(w_i \mid v_j)$  being calculated differently. The raw probability of a category given a lists of words in the test document was derived from  $Prior(v_j)$  \* the product of  $\frac{w_i instances \ count \ in \ v_j}{total \ word \ count \ in \ v_j}$  for each w k in test document.

The M-Est probability was derived from  $Prior(v_j)$  \* the product of  $\frac{w_i \ln stances \ count \ in \ v_j + 1}{total \ word \ count \ in \ v_j + total \ count \ unique \ words \ seen \ in \ train.txt}$  for each w\_k in test document.

By adding one to the numerator, the program decides not to rule out the possibility of w\_k appearing in test document v\_j even though it did not appear in v\_j for the training data. By adding the total count of unique words seen in train.txt to the denominator, the program is able to make the probability of a new word appearing in v\_j very small, but not zero because of the previously mentioned 1 addition.

However, the tf-idf version was by far the best. This is due to weighing the w\_k frequency[tf] (of a given word position in test document) to the inverse document frequency[idf].

The term frequency is a measure of the probability of a word given a category, as seen in fraction form in for the

analysis of raw probability. However, this tf gets multiplied by the idf before being multiplied into the total product.

The idf measures the significance of a word by computing  $\log(\frac{total\ number\ of\ documents\ in\ train\ test}{number\ of\ documents\ with\ w_k})$ 

Then, by multiplying the tf and idf, we get a weighted probability of w\_k in v\_j. For each w\_k in test document v\_j, the product of idf\*tf gets multiplied by the previous products of w\_k weighted probability. If there are no previous values, then the given value becomes the first.

By weighing the significance of a word to its probability, tfidf distinguishes itself from the other categories as the most successful at predicting categories.